MULTIMEDIA UNIVERSITY®

STUDENT ID NO

MULTIMEDIA UNIVERSITY

FINAL EXAMINATION

TRIMESTER 1, 2016/2017

TDP 3471 – DISTRIBUTED AND PARALLEL COMPUTING

(All sections / Groups)

22nd OCTOBER 2016
9.00 a.m - 11.00 a.m
(2 Hours)

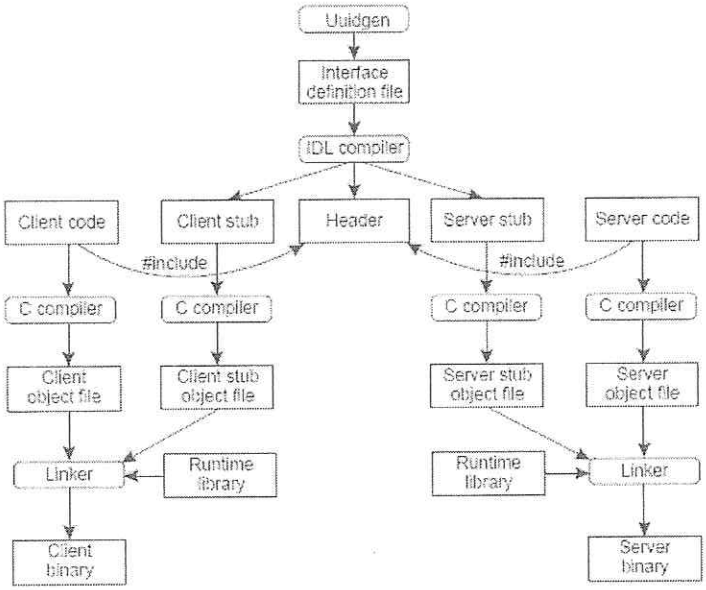
INSTRUCTIONS TO STUDENTS

1. This question paper consists of 6 pages, including this page, with 4 questions.
2. Attempt **ALL** questions. All questions carry equal marks and the distribution of the marks for each question is given.
3. Please print all your answers in the Answer Booklet provided.

Question 1 [10 marks]

Q1a Referring to the figure below, explain which functions are implemented in the runtime library of an RPC system.

(2 marks)



Q1b Explain the principle workings of a remote procedure call (RPC).

(2 marks)

Q1c Traditional RPC mechanisms cannot handle pointers. What problem will this create and how can it be addressed?

(4 marks)

Q1d Explain what ***relocation transparency*** is. Provide a concrete example on how ***relocation transparency*** is commonly applied in a Distributed System.

(2 marks)

Continued...

Question 2 [10 marks]

Q2a Based on your understanding, describe the function of a network socket. Briefly explain the differences between UDP Datagram Communication and TCP Stream Communication.

(4 marks)

Q2b To provide fault tolerance, it is critical to determine how much replication is needed when using process groups that are able to tolerate faults with respect to this, briefly explain what is a k fault-tolerant group, and how does k depend on failure semantics?

(3 marks)

Q2c Explain your answer with help of a diagram, that having three processes of which one is faulty, is not enough to guarantee agreement between the two non-faulty ones in a Byzantine agreement problem setting.

(3 marks)

Continued...

Question 3 [10 marks]

Q3a Briefly describe ONE advantage and ONE disadvantage of Bittorrent file sharing protocol.

(2 marks)

Q3b Briefly discuss the differences between the Clustered Computing and Distributed Processing.

(4 marks)

Q3c Explain with help of a diagram the differences between blocking and non- blocking message passing and how is it different in MPI.

(4 marks)

Continued...

Question 4 [10 marks]

Q4a Suppose that MPI COMM WORLD consists of the three processes 0,1, and 2, and the following code is executed:

```
int x, y, z;
switch(my_rank) {
case 0: x=0; y=1; z=2;
MPI_Bcast(&x, 1, MPI_INT, 0, MPI_COMM_WORLD);
MPI_Send(&y, 1, MPI_INT, 2, 43, MPI_COMM_WORLD);
MPI_Bcast(&z, 1, MPI_INT, 1, MPI_COMM_WORLD);
break;
case 1: x=3; y=8; z=5;
MPI_Bcast(&x, 1, MPI_INT, 0, MPI_COMM_WORLD);
MPI_Bcast(&y, 1, MPI_INT, 1, MPI_COMM_WORLD);
break;
case 2: x=6; y=7; z=8;
MPI_Bcast(&z, 1, MPI_INT, 0, MPI_COMM_WORLD);
MPI_Recv(&x, 1, MPI_INT, 0, 43, MPI_COMM_WORLD, &status);
MPI_Bcast(&y, 1, MPI_INT, 1, MPI_COMM_WORLD);
break;
}
```

What are the values of x, y, and z on each process after the code has been executed?

(3 marks)

Q4b What is superlinear speedup? Suggest two ways/approaches to achieve superlinear speedup in terms of parallel program application and processing element (hardware).

(2 marks)

Q4c Suppose that 5% of a program's code cannot be parallelized at all and the remaining 95% can be perfectly parallelized:

(i) If we use 100 processors to run this program in parallel, what would be the maximum speedup that can be obtained according to Amdahl's law?
(ii) Calculate the efficiency for the case in (i).
(iii) What is the scaled speedup factor according to Gustafson's law?
(iv) What is the efficiency for the case in (iii)?

(2 marks)

Continued...

Q4d Given the following MPI program, answer the following questions.

```
#include <stdio.h>
#include <stdlib.h>
#include "mpi.h"
void main(int argc, char** argv)
{
int my_id,i,p;
double a,t,s;
MPI_Status status;
MPI_Init(&argc,&argv);
MPI_Comm_rank(MPI_COMM_WORLD,&my_id);
MPI_Comm_size(MPI_COMM_WORLD,&p);

if (my_id==0) {
      a=rand()%6;
      for (i=0;i<p;i++)
            MPI_Send(&a,1,MPI_DOUBLE,i,0,MPI_COMM_WORLD);
}
            MPI_Recv(&a,1,MPI_DOUBLE,0,0,MPI_COMM_WORLD,&status);

if (my_id==0) {
      s=a;
      t=a;
      MPI_Send(&s,1,MPI_DOUBLE,1,0,MPI_COMM_WORLD);
      MPI_Send(&t,1,MPI_DOUBLE,1,0,MPI_COMM_WORLD);
      }

      if (my_id>0) {
      MPI_Recv(&s,1,MPI_DOUBLE,my_id-1,0,MPI_COMM_WORLD,&status);
      MPI_Recv(&t,1,MPI_DOUBLE,my_id-1,0,MPI_COMM_WORLD,&status);
      t=t*a;
      s=s+t/(my_id+1);
      MPI_Send(&s,1,MPI_DOUBLE,(my_id+1)%p,0,MPI_COMM_WORLD);
      MPI_Send(&t,1,MPI_DOUBLE,(my_id+1)%p,0,MPI_COMM_WORLD);
      }

      if (my_id==0) {
            MPI_Recv(&s,1,MPI_DOUBLE, p-1,0,MPI_COMM_WORLD,&status);
            printf("a=%f and  s=%f\n",a,s);
      }
      MPI_Finalize();
}
```

(i) Briefly explain what the program above does. (1 mark)

(ii) Give the output of the program when a = 5 and 4 processes are used. (2 marks)

END OF EXAM PAPER